# ГЛАВА XZ.
# Скоринговые функции ошибки

*... точные утверждения теряют значимость,*
*а значимые утверждения теряют точность.*
Л. Заде

*В жизни нет гарантий,*
*существуют одни вероятности.*
Т. Клэнси

В этой главе рассматриваем задачу классификации на два класса с метками из множества $\{0,1\}$, при этом алгоритм должен выдавать оценку принадлежности к классу 1, т.е. значение из отрезка $[0,1]$. Заметим, что если алгоритм выдал оценку 1, т.е. максимально возможную, то выданное значение формально совпадает с меткой класса 1, в которой алгоритм уверен. Аналогично, если алгоритм выдал оценку 0, то алгоритм уверен, что объект принадлежит классу 0. Также возможны и промежуточные значения, например 0.7.

Заметьте, что (пока) алгоритм выдаёт «оценку», а не вероятность. Вероятность ли это – отдельный вопрос.

Функция ошибки $L(a,y)$ однозначно задаётся здесь двумя функциями: $L(a,1)$ (штраф для объекта из класса 1) и $L(a,0)$ (штраф для объекта из класса 0):

$$L(a,y)=\begin{cases} L(a,1), & y=1, \\ L(a,0), & y=0, \end{cases}$$

причём часто $L(a,1)=L(1-a,0)$ (штрафы симметричны[1]), такое представление функции ошибки назовём **«раздельной формой записи»**. Часто используют более сложную для понимания (но удобную для использования и запоминания) **«совместную форму записи»**:

$$L(a,y)=yL(a,1)+(1-y)L(a,0).$$

[1] Например, штраф за ответ 0.7 при истинной метке 1 совпадает со штрафом за ответ 0.3 при метке 0.

## Логистическая функция ошибки

Первая популярная функция ошибки в описанной выше задаче – **логистическая** или **logloss**, её называют ещё перекрёстной или кросс-энтропией (Cross Entropy), а также ошибкой Кульбака-Лейблера (Kullback-Leibler loss):

$$\text{logloss} = -\frac{1}{m}\sum_{i=1}^{m}(y_i \log a_i + (1-y_i)\log(1-a_i)). \qquad \text{(XZ.1)}$$

(как и раньше, считаем, что объекты выборки $x_1, x_2, \ldots, x_m$ имеют известные метки $y_1, y_2, \ldots, y_m$ и для них получены оценки $a_1, a_2, \ldots, a_m$). Если от совместной формы записи перейти к раздельной, то $i$-е слагаемое в указанной сумме выглядит достаточно просто:

$$-\begin{cases} \log a_i, & y_i = 1, \\ \log(1-a_i), & y_i = 0. \end{cases}$$

По графикам приведённых функций, см. рис. XZ.1, становится понятно, что такая функция ошибок сильно наказывает за неправильные ответы: если вместо 0 выдать 1 (или наоборот), то ошибка равна бесконечности ($+\infty$). При правильном ответе ошибка нулевая.

Практически в любой реализации максимальный log_loss-штраф равен некоторой большой константе.

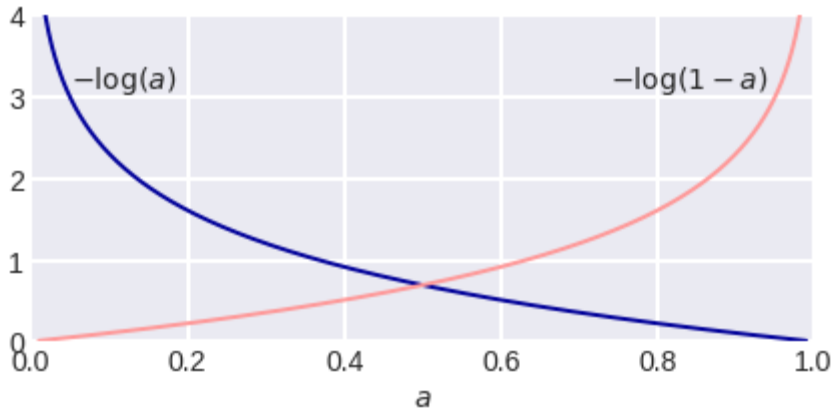


Рис. XZ.1. Логистическая ошибка ответа $a$ на одном объекте.

Выведем оптимальный константный алгоритм для выборки из $m_1$ представителей класса 1 и $m_0$ представителей класса 0, $m = m_0 + m_1$. Необходимо решить следующую задачу оптимизации:

$$-\frac{1}{m}\sum_{i=1}^{m}(y_i \log a + (1-y_i)\log(1-a)) \to \min_a$$

или (в сумме $m_1$ слагаемых вида $\log a$ и $m_0$ вида $\log(1-a)$)

$$-\frac{m_1}{m}\log a - \frac{m_0}{m}\log(1-a) \to \min_a .$$

Если взять производную (по $a$) и приравнять к нулю, то получим

$$-\frac{m_1}{m}\frac{1}{a} + \frac{m_0}{m}\frac{1}{1-a} = 0$$

или

$$a = \frac{m_1}{m},$$

т.е. оптимальная константа равна доле объектов класса 1 в выборке. На рис. XZ.2 изображён график logloss-ошибки константного алгоритма для выборки из четырёх объектов класса 0 и 6 объектов класса 1.

Оптимальный для logloss константный алгоритм выдаёт долю объектов класса 1.

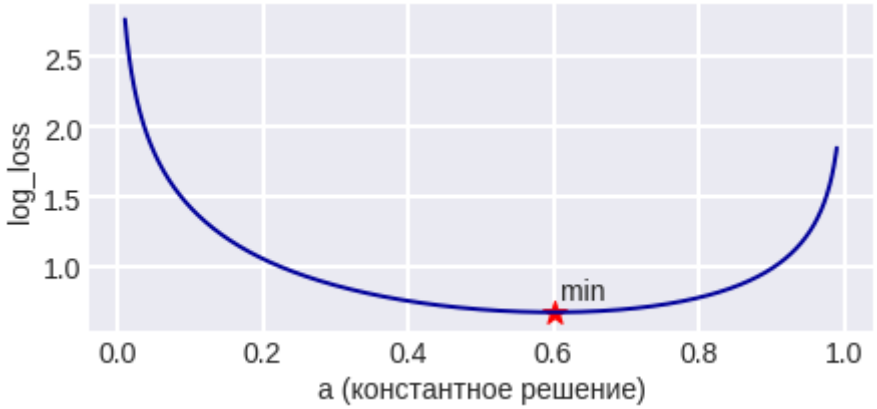


Рис. XZ.2. График logloss константного решения для выборки, в которой 60% объектов класса 1.

Пусть для некоторого $i$-го объекта известна вероятность $p$ принадлежности к классу 1, посмотрим, какой ответ оптимален на этом объекте с точки зрения минимизации ошибки logloss, точнее матожидания ошибки. Запишем матожидание ошибки

$$-p\log(a_i) - (1-p)\log(1-a_i)$$

и минимизируем его: возьмём производную и приравняем её к нулю, получаем

$$\frac{p}{a_i} - \frac{1-p}{1-a_i} = 0,$$

$$a_i = p.$$

Таким образом, оптимально для каждого объекта выдавать его вероятность принадлежности к классу 1, т.е. **для минимизации logloss надо уметь корректно вычислять (оценивать) вероятности принадлежности к классам!** Если подставить полученное оптимальное решение в минимизируемый функционал, то получим энтропию:

Оптимизация logloss «заставляет» алгоритм хорошо оценивать вероятности принадлежности к классам.

$$-p\log(p) - (1-p)\log(1-p)$$

(напомним, что это минимальное матожидание logloss-ошибки на объекте с известной вероятностью $p$ принадлежности классу 1). Это объясняет, почему при построении решающих деревьев[1] в задачах классификации применяют энтропийный критерий расщепления (ветвления). Дело в том, что оценка принадлежности к классу 1 в дереве часто формируется с помощью среднего арифметического меток в листе (т.е. с помощью оптимальной оценки вероятности). В любом случае, в конкретном дереве эта оценка вероятности будет одинакова для всех объектов в листе, т.е. будет константой, а энтропия в листе равна оценке (сверху) logloss-ошибки константного решения. **Используя энтропийный критерий, мы неявно оптимизируем логлосс!**

Минимальное значение ошибки logloss равно 0, максимальное – $+\infty$, но «эффективным максимальным» можно считать ошибку при использовании оптимального константного алгоритма (вряд же в итоге решения получится алгоритм хуже константного), т.е.

$$\left[0, -\frac{m_1}{m}\log\frac{m_1}{m} - \frac{m_0}{m}\log\frac{m_0}{m}\right].$$

Интересно, что если брать логарифм по основанию 2, то на сбалансированной выборке этот отрезок «превращается» в

Как правило, $\text{logloss} \in [0,1]$.

[1] Также при построении случайных лесов и, иногда, деревьев в бустингах.

отрезок [0, 1].

Дадим **теоретическое обоснование логистической ошибки**. Обучающую выборку (точнее, метки в ней) в задаче бинарной классификации можно рассматривать, как реализацию обобщённой схемы Бернулли: для каждого объекта генерируется метка – случайная величина, которая с вероятностью $p$ (своей для каждого объекта) принимает значение 1 и с вероятностью $(1-p)$ – значение 0. Предположим, что модель как раз и должна генерировать правильные вероятности, тогда можно записать функцию правдоподобия:

$$p(y \mid X, w) = \prod_i p(y_i \mid x_i, w) = \prod_i a_i^{y_i} (1-a_i)^{1-y_i},$$

где $a_i = a(x_i \mid w)$ – ответ алгоритма, зависящего от параметров $w$, на $i$-м объекте (произведение проводится по всем номерам объектов выборки). Если воспользоваться методом максимального правдоподобия, то получим следующую задачу оптимизации:

$$\log p(y \mid X, w) = \sum_i (y_i \log a_i + (1-y_i)\log(1-a_i)) \to \max,$$

которая эквивалентна минимизации логистической ошибки (XZ.1) на нашей выборке:

$$\sum_i (-y_i \log a_i - (1-y_i)\log(1-a_i)) \to \min$$

(нормировочный множитель не влияет на решение задачи оптимизации). Для задачи бинарной классификации, в которой алгоритм должен выдать вероятность принадлежности классу 1, logloss-ошибка логична ровно настолько, насколько логична MSE в задаче линейной регрессии с нормальным шумом (поскольку **обе функции ошибки выводятся из метода максимального правдоподобия**). Заметим, что в данном случае мы не предполагали, что наш алгоритм принадлежит какой-то определённой модели.

Метод максимального правдоподобия неявно «пронизывает» всё машинное обучение.

Во многих книгах логистической функцией ошибки («logistic loss») называется выражение, отличное от (XZ.1), которое мы сейчас получим, подставив выражение для сигмоиды

$$\sigma(w^{\mathrm{T}} x) \equiv \frac{1}{1+e^{-w^{\mathrm{T}} x}}$$

в logloss и сделав переобозначение: считаем, что метки классов теперь –1 и +1, тогда

$$\text{logloss}(a,y)=\begin{cases}-\log a, & y=+1,\\ -\log(1-a), & y=-1,\end{cases}=\begin{cases}-\log\left(\dfrac{1}{1+\exp(-w^{\mathrm{T}}x)}\right), & y=+1,\\ -\log\left(\dfrac{\exp(-w^{\mathrm{T}}x)}{1+\exp(-w^{\mathrm{T}}x)}\right), & y=-1,\end{cases}=$$

$$=\begin{cases}-\log\left(\dfrac{1}{1+\exp(-w^{\mathrm{T}}x)}\right), & y=+1,\\ -\log\left(\dfrac{1}{\exp(+w^{\mathrm{T}}x)+1}\right), & y=-1,\end{cases}=\begin{cases}\log\left(1+\exp(-w^{\mathrm{T}}x)\right), & y=+1,\\ \log\left(1+\exp(+w^{\mathrm{T}}x)\right), & y=-1.\end{cases}$$

В результате получили, что

$$\text{logloss}(\sigma(w^{\mathrm{T}}x),y)=\log(1+\exp(-y\cdot w^{\mathrm{T}}x)). \qquad \text{(XZ.2)}$$

Полезно посмотреть на график функции $\log(1+\exp(z))$, которая является сглаженным аналогом функции[1] $\max[0,x]$, см. рис. XZ.3. Если при настройке весов линейного классификатора минимизировать logloss, то таким образом мы обучаем классическую логистическую регрессию, если же использовать чуть подправленную функцию $\max[0,x]$ – т.н. Hinge loss и добавить регуляризацию, то получаем обучение метода опорных векторов (SVM):

$$\sum_i \max[1-y_i w^{\mathrm{T}}x,\,0]+\alpha\cdot w^{\mathrm{T}}w\to\min.$$

При обучении метода релевантных векторов (RVM, Relevance Vector Machine) задача оптимизации похожа на рассмотренные выше:

$$\sum_i \log(1+\exp(-y_i w^{\mathrm{T}}x))+w^{\mathrm{T}}\,\mathrm{diag}(\alpha)w\to\min.$$

Часто разные методы можно получать друг из друга «немного подправив» оптимизируемые функции.

Покажем теперь связь логистической функции ошибки с **расхождением Кульбака-Лейблера и перекрёстной энтропией**. Известно, что перекрёстная энтропия (cross entropy) для двух распределений вводится как

$$H(P,Q)=-\int p(z)\log q(z)\,\partial z,$$

[1] В глубоком обучении такую функцию принято называть ReLU (Rectified Linear Unit).

где $P$ и $Q$ – распределения, $p$ и $q$ – плотности этих распределений, а для дискретных распределений интеграл заменяется на сумму:

$$H(P,Q) = -\sum_i P_i \log Q_i ,$$

где $P_i$, $Q_i$ – вероятности дискретных событий. Рассмотрим конкретный объект $x$ с меткой $y$. Если алгоритм выдаёт вероятность $a$ принадлежности первому классу, то предполагаемое распределение на событиях «класс 0», «класс 1» – $(1-a, a)$, а истинное – $(1-y, y)$. Запишем для этих распределений перекрёстную энтропию:

logloss = cross entropy на распределениях (1 – a, a), (1 – y, y).

$$-(1-y)\log(1-a) - y\log a ,$$

она как раз совпадает с логистической функцией ошибки.

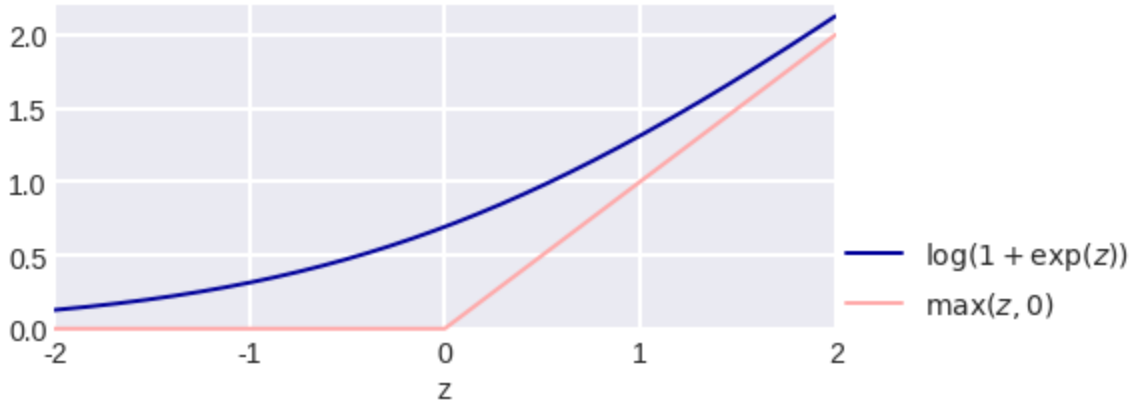


Рис. XZ.3. Графики тождественной функции и $\log(1+\exp(z))$.

Расхождение[1] (дивергенцию) Кульбака-Лейблера (KL, Kullback–Leibler divergence) часто используют (особенно в машинном обучении, байесовском подходе и теории информации) для вычисления непохожести двух распределений. Оно определяется по следующей формуле:

$$D_{\mathrm{KL}}(P \,\|\, Q) = \int p(z) \log \frac{p(z)}{q(z)} \partial z = H(P,Q) - H(P) ,$$

где $P$ и $Q$ – распределения (первое обычно «истинное», а второе – то, про которое нам интересно, насколько оно похоже на истинное), $p$ и $q$ – плотности этих распределений. В данном случае энтропия

[1] KL-расхождение называют «расстоянием», хотя оно не является симметричным и не удовлетворяет неравенству треугольника.

$$H(P) = -\int p(z)\log p(z)\partial z$$

истинного распределения $(1-y, y)$ равна нулю, так как значение

$$-(1-y)\log(1-y) - y\log y$$

считается нулевым[1]. Поэтому расхождение Кульбака-Лейблера в задаче бинарной классификации с чёткими истинными метками в точности совпадает с логистической функцией ошибки.

## Скоринговые ошибки

Логистическая функция ошибки в задаче бинарной классификации обладает «правильным свойством»: оптимальной ответ на каждом объекте – вероятность его принадлежности классу 1. Подобные функции ошибки называются **скоринговыми (proper scoring rules[2])**, т.е. функции $L(y,a)$, для которых

$$p = \arg\min_a \mathbf{E}_y L(y,a), \text{ для } y \sim \text{Bernoulli}(p).$$

Покажем, что есть и другие скоринговые функции. Рассмотрим ошибку MSE (Mean Squared Error) в совместной форме записи[3]:

$$\text{SE} = (y-a)^2 = y(1-a)^2 + (1-y)a^2.$$

Если объект с вероятностью $p$ принадлежит классу 1, то матожидание ошибки

$$p(1-a)^2 + (1-p)a^2,$$

Совместная форма записи удобна тем, что часто матожидание ошибки получается заменой *y* на *p*.

при её минимизации получаем (берём производную и приравниваем к нулю):

$$-2p(1-a) + 2(1-p)a = 0,$$

$$a = p,$$

[1] 0·log0 = 0

[2] Формально, переводя с английского, корректнее называть описанную функцию ошибки «правильной / подходящей скоринговой», но мы упростим терминологию.
Buja A., Stuetzle W., Shen Y. Loss functions for binary class probability estimation and classification: Structure and applications //Working draft, November. – 2005. – Т. 3. – С. 13.

[3] Здесь второе равенство не верно в общем случае, но выполняется для $y \in \{0,1\}$.

т.е. SE – тоже скоринговая функция ошибки. Интересно, что минимум матождания при этом:

$$p(1-p)^2 + (1-p)p^2 = (1-p)p.$$

Это выражение с точностью до мультипликативной константы 2 совпадает с функцией, которую оптимизируют при построении решающих деревьев в критерии Джини. Функцию ошибки MSE в задачах классификации называют **«ошибкой Брайера» (Brier score[1])**, именно под таким названием она реализована в scikit-learn[2]. Если вспомнить аналогичный вывод для логистической функции ошибки, то там матожидание функции ошибки являлось энтропией и соответствующий критерий расщепления был энтропийный. Таким образом, выбор критерия расщепления среди двух стандартных (энтропийного и Джини) – это выбор между целевыми функциями для оптимизации[3].

На самом деле, любая скоринговая функция порождает информационную меру, которая может быть использована в критерии расщепления.

Рассмотрим теперь такую функцию ошибки – **misclassification loss (error)**:

$$\text{ME} = y\,I[a \le 0.5] + (1-y)\,I[a > 0.5],$$

тогда

$$\mathbf{E}_y \text{ME} = p\,I[a \le 0.5] + (1-p)\,I[a > 0.5],\ y \sim \text{Bernoully}(p),$$

и задача минимизации матожидания ошибки не имеет единственного решения:

$$\arg\min \mathbf{E}_y \text{ME} \in \begin{cases} [0, 0.5], & p \le 0.5, \\ [0.5, 1], & p > 0.5, \end{cases}$$

т.е. подходит любое решение, которое при округлении даёт верную метку, например для целевого вектора (0, 1, 0, 0) верным будет решение (0.4, 0.6, 0, 0.1), при этом минимум матожидания ошибки равен

$$\min \mathbf{E}_y \text{ME} = \min(p, 1-p).$$

[1] Brier G. W. Verification of forecasts expressed in terms of probability //Monthly weather review. – 1950. – Т. 78. – №. 1. – С. 1-3.

[2] sklearn.metrics.brier_score_loss

[3] В различных форумах часто мелькают мнения, что MSE-ошибку нельзя использовать в задачах классификации. Оказывается, что можно, более того, иногда она даже предпочтительнее logloss-а. Детали можно найти в этой статье: http://www.eecs.harvard.edu/cs286r/courses/fall12/papers/Selten98.pdf

На самом деле, описанная функция ошибки немного «фиктивная». В ней не важно значение ответа нашего алгоритма, гораздо важнее: перевалило ли оно порог 0.5. Оптимизация подобной функции может не привести к тому, что ответы алгоритма разумно интерпретировать как вероятности[1].

Теперь давайте в логистической функции заменим логарифм на очень похожую (см. рис. XZ.4) функцию

$$\sqrt{\frac{1-a}{a}}.$$

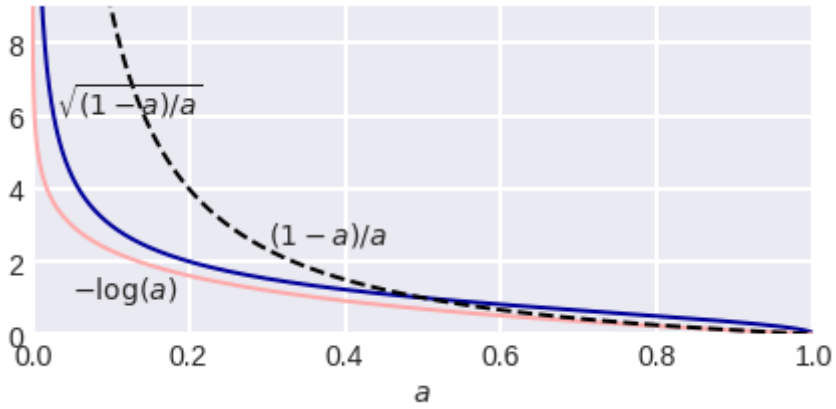


Рис. XZ.4. Несколько похожих функций.

Получим такое выражение (в совместной форме записи):

$$\text{exploss} = y\sqrt{\frac{1-a}{a}} + (1-y)\sqrt{\frac{a}{1-a}}.$$

Проделаем стандартную процедуру: минимизируем матожидание ошибки для объекта, который с вероятностью $p$ принадлежит классу 1:

$$p\sqrt{\frac{1-a}{a}} + (1-p)\sqrt{\frac{a}{1-a}} \to \min.$$

Заметим, что

$$\frac{\partial}{\partial a}\sqrt{\frac{1-a}{a}} = -\frac{1}{2a^2}\sqrt{\frac{a}{1-a}},$$

[1] Однако вероятности минимизируют матожидание ME, такие функции называют **нестрогими скоринговыми (inproper scoring rules)**.

$$\frac{\partial}{\partial a}\sqrt{\frac{a}{1-a}}=\frac{1}{2(1-a)^2}\sqrt{\frac{1-a}{a}}\,.$$

Поэтому, приравняв к нулю производную матожидания, получаем (цветом обозначено домножение частей равенства на общий ненулевой множитель)

$$-\frac{p}{2a^2}\sqrt{\frac{a}{1-a}}+\frac{1-p}{2(1-a)^2}\sqrt{\frac{1-a}{a}}=0,$$

$$\frac{1-p}{2(1-a)^2}\sqrt{\frac{1-a}{a}}\sqrt{\frac{1-a}{a}}=-\frac{p}{2a^2}\sqrt{\frac{a}{1-a}}\sqrt{\frac{1-a}{a}},$$

$$\frac{1-p}{1-a}=\frac{p}{a},$$

$$a=p\,.$$

Получили ещё одну скоринговую функция ошибки. Минимальное матожидание для функции exploss равно

$$p\sqrt{\frac{1-p}{p}}+(1-p)\sqrt{\frac{p}{1-p}}=2\sqrt{p(1-p)}$$

(т.е. корень из минимума матожидания при использовании ME).

Функция exploss имеет такое название по следующей причине[1]. Рассмотрим задачу классификации на два класса $\{\pm 1\}$ и алгоритм $a(x)$ для её решения, который выдаёт некоторые оценки принадлежности к классу 1 из интервала $(-\infty,+\infty)$: чем больше оценка, тем «более уверен» алгоритм в метке 1. Очень логичной кажется такая функция ошибки:

$$\exp(-y\,a),$$

которая изначально использовалась в бустинге: ошибка всегда ненулевая, но если мы ошибаемся в знаке ($y\,a<0$, например выдаём отрицательную оценку при метке $y=+1$), то ошибка экспоненциально возрастает при росте уверенности в неправильном ответе ($|\,a\,|\to+\infty$). Матожидание такой ошибки на объекте, который с вероятностью $p$ принадлежит классу 1:

[1] В её определении не используется экспонента, что может вызывать удивление.

$$p\exp(-a)+(1-p)\exp(+a),$$

если взять производную и приравнять к нулю, то получим,

$$\frac{\partial}{\partial a}\left[p\exp(-a)+(1-p)\exp(+a)\right]=0,$$

$$-p\exp(-a)+(1-p)\exp(+a)=0,$$

$$a=\ln\sqrt{\frac{p}{1-p}}.$$

Этот приём называется переводом ответов в вероятностную шкалу (probability scale).

Если подставить это выражение в исходную функцию ошибки, то получим как раз выражение exploss (только вместо ответов алгоритма там стоит вероятность):

$$a=\exp\left(-y\ln\sqrt{\frac{p}{1-p}}\right)=\left(\frac{1-p}{p}\right)^{y/2}.$$

(помним также, что в этом выражении $y\in\{\pm1\}$). Таким образом, введённый выше exploss это «естественная поправка» экспоненциальной ошибки $\exp(-ya)$ на случай, когда мы хотим получать не произвольные вещественные оценки, а оценки из отрезка $[0,1]$ и интерпретировать их как вероятности.

Приведём также несколько примеров нескоринговых функций. Если из выражения для exploss убрать корни, т.е. прийти к выражению

$$y\frac{1-a}{a}+(1-y)\frac{a}{1-a},$$

то уже не получим скоринговой функции[1], а функция $(1-a)/a$ не так похожа на логарифм, см. рис. XZ.4[2]. Также не является скоринговой функция MAE (Mean absolute error), в которой ошибка на объекте

$$\mathrm{AE}=|y-a|=y(1-a)+(1-y)a,$$

здесь учитываем, что $a\in[0,1]$. Минимизация математического ожидания ошибки приводит к такой задаче

[1] Проверку оставляем читателю, см. задание.

[2] Хотя это, конечно, не является аргументом в пользу её нескоринговости.

$$p(1-a)+(1-p)a = p+a-2pa = 2a\left(\frac{1}{2}-p\right)+p \to \min .$$

Таким образом,

$$a = \text{round}(p) \equiv \begin{cases} 1, & p \geq 0.5, \\ 0, & p < 0.5, \end{cases}$$

т.е. оптимальный ответ не совпадает с вероятностью. Более того, здесь выгодно давать только ответы 0 или 1. При этом минимальное матожидание равно[1]

Нескоринговые функции нельзя использовать при настройке алгоритма, если вы хотите ответ интерпретировать как вероятность.

$$\min(p, 1-p).$$

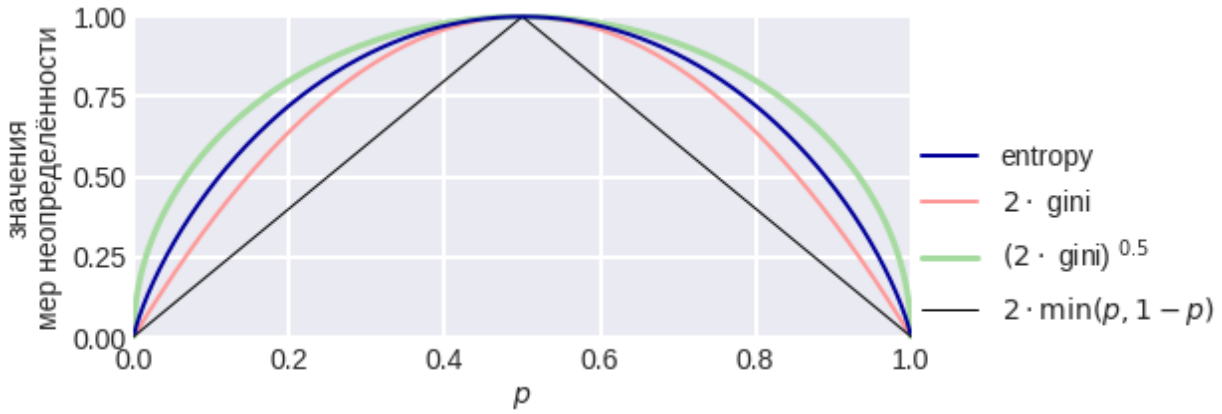


Рис. XZ.5. Различные меры неопределённости в задаче бинарной классификации.

На рис. XZ.5 показаны меры неопределённости в задаче бинарной классификации, мы установили следующее соответствие (см. табл. – из неё понятно, как критерии расщепления в решающих деревьях связаны с функциями ошибки). Интересно, что график энтропии лежит между графиками нормированных функций Джини и её корня.

[1] Заметим, что мы уже встречали такое выражение.

| функция ошибки | мера неопределённости | |
|---|---|---|
| LogLoss | Энтропия | $-p\log(p)-(1-p)\log(1-p)$ |
| MSE (Brier Score) | Джини | $1-p^2-(1-p)^2=2p(1-p)$ |
| ExpLoss | | $2\sqrt{p(1-p)}$ |
| ME (Missclassification error) | MC (Missclassification criteria) | $\min(p,1-p)$ |

## Приложения и примеры

1. В одном из реальных проектов в задаче бинарной классификации заказчик предложил следующую функцию ошибки:

$$L(a,y)=|y-a|\cdot\begin{cases}0.8, & y=1,\\ 0.2, & y=0,\end{cases}$$

где $y\in\{0,1\}$ – верная классификация $i$-го объекта, $a\in[0,1]$ – ответ нашего алгоритма. При этом заказчик подчёркивал, что ему важно, чтобы алгоритм получал промежуточные значения из интервала $(0,1)$. Проведём анализ такой функции ошибки стандартным методом, который мы применяли в этой главе. Найдём матожидание ошибки для объекта, который с вероятностью $p$ принадлежит классу 1:

$$0.8|1-a|\,p+0.2|a|(1-p)=$$

$$=0.8p-0.8pa+0.2a-0.2pa=$$

$$=0.8p-(p-0.2)a$$

(здесь при раскрытии модуля учитываем, что $a\in[0,1]$). Оптимальное решение (которое минимизирует матожидание ошибки)

$$a=\begin{cases}0, & p<0.2,\\ 1, & p\geq 0.2,\end{cases} \qquad \text{(XZ.3)}$$

(см. также рис. XZ.6).

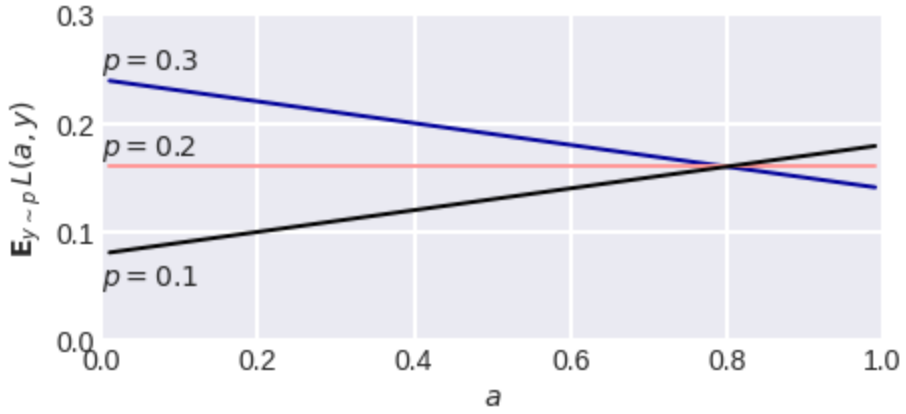


Рис. XZ.6. Ошибка при разных вероятностях принадлежности к классу 1.

Таким образом, функция ошибки вынуждает нас выдавать ответы из множества {0,1}, а не из внутренности отрезка [0, 1], поэтому требования заказчика противоречивы (если мы действительно хотим минимизировать указанную ошибку). Интересен график величины ошибки оптимального решения в зависимости от вероятности $p$. При минимальной и максимальной вероятности ошибка минимальна, а максимальна при $p = 0.2$.

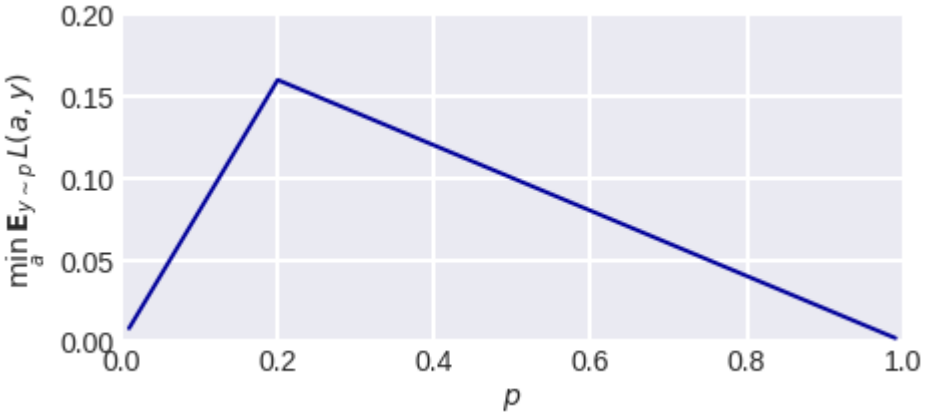


Рис. XZ.7. Минимальное матожидание ошибки.

2. В продолжение предыдущего примера отметим, что для каждого объекта можно оценить вероятность принадлежности к классу 1, а затем (для получения окончательного ответа) её бинаризовать по правилу (XZ.3), чтобы минимизировать заданный функционал.

В принципе, это практически универсальный подход к решению задач классификации. Если мы знаем вероятность принадлежности объекта разным классам, то легко его классифицировать. Байесовский классификатор как раз это и делает, являясь оптимальным классификатором. Но надо учитывать, что

- вероятности и плотности оцениваются с некоторой погрешностью,

- надо минимизировать заданную функцию ошибки,
- необходимости знать вероятности может не быть.

Последние два замечания делают машинное обучение самостоятельной дисциплиной. Если бы решение любой задачи сводилось к оцениванию вероятностей и плотностей, то она бы была разделом теории вероятностей и математической статистики.

3. При обучении логистической регрессии как раз минимизируется ошибка (XZ.2). Если взять производную logloss, то получим

$$\frac{\partial \text{logloss}(a,y)}{\partial w} = \frac{1}{1+\exp(-y\cdot w^{\mathrm{T}}x)}\exp(-y\cdot w^{\mathrm{T}}x)\cdot(-y\cdot x) =$$

$$= \begin{cases} -\dfrac{1}{1+\exp(w^{\mathrm{T}}x)}x, & y=+1, \\ +\dfrac{1}{1+\exp(-w^{\mathrm{T}}x)}x, & y=-1. \end{cases}$$

Если вернуться к меткам $y\in\{0,1\}$, тогда получаем

$$\begin{cases} -(1-\sigma(w^{\mathrm{T}}x))\cdot x, & y=1, \\ \sigma(w^{\mathrm{T}}x)\cdot x, & y=0. \end{cases}$$

Поэтому шаг стохастического градиентного спуска (на одном объекте) запишется в виде

$$w \leftarrow w+\eta\cdot(y-\sigma(w^{\mathrm{T}}x))\cdot x\,.$$

В скобке оценивается различие метки и ответа алгоритма, чем оно больше, тем сильнее корректируются веса. Формула очень похожа на аналогичную для обучения линейной регрессии.

4. Для оптимизации скоринговых функций применяют калибровку уверенности. Подробнее о ней поговорим в главе о постобработке ответов. В рассматриваемой здесь задаче бинарной классификации калибровка заключается в построении некоторого отображения $c:\mathbb{R}\to[0,1]$[1] (т.н. «деформации»), которое корректирует первоначально полученную оценку $b(x)$ принадлежности к классу 1 объекта $x$:

[1] В общем случае первоначальная оценка может не лежать на отрезке [0, 1].

$$a(x) = c(b(x)) .$$

Например, в калибровке Платта (Platt calibration[1]) используется функция деформации в виде сигмоиды $c(z) = \sigma(\alpha x + \beta)$ с вещественными параметрами $\alpha$, $\beta$, которые можно настроить минимизируя заданную функцию ошибки на отложенной выборке.

[1] Platt, John et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Advances in large margin classifiers, 10(3): 61–74, 1999.

## Задачи и вопросы

1. При каких степенях $q$ функция ошибки

$$y\left(\frac{1-a}{a}\right)^q + (1-y)\left(\frac{a}{1-a}\right)^q$$

является скоринговой?

2. Является ли скоринговой функция[1] $(1+y-a)\cdot\exp(a)$?

3. Рассмотрим в задаче бинарной классификации информационную меру $\min(p,1-p)$. Каким функциям ошибки она соответствует (есть несколько ответов)? Являются ли они скоринговыми?

4. Рассмотрите задачу регрессии, в которой алгоритм должен выдать оценку «распределения на вещественной оси целевых значений». Как здесь построить теорию скоринговой оценки? Можно ли каким-то искусственным приёмом строить скоринговые оценки в регрессии (подсказка: специальным сведением к задачам классификации)?

5. Как построить диалог с заказчиков в примере 1? Очевидно, что он хотел сильнее штрафовать за отнесение к классу 1, при этом получать оценки, которые можно интерпретировать как вероятности. Возможно, абсолютные отклонения оценок от меток для него понятны и имеют какую-то бизнес-интерпретацию. Можно ли удовлетворить его требования? Почему минимум матожидания ошибки максимален при $p=0.2$? Чему равен этот максимум и как проинтерпретировать это значение?

6. Выведите формулу для коррекции весов логистической регрессии методом стохастического градиентного спуска при минимизации ошибки Брайера. Проинтерпретируйте её, будут ли проблемы со сходимостью при использовании этой формулы?

[1] Vince’s crazy proper scoring rule,. https://www2.cs.duke.edu/courses/spring17/compsci590.2/proper_scoring.pdf

## Скоринговые функции ошибки: итоги

1. Анализ матожидания ошибки помогает много понять о ней. Матожидание часто легко выписывается по раздельной форме записи функции ошибки.

2. Минимизация скоринговых функций ошибки «заставляет» алгоритм выдавать хорошие оценки принадлежности к классам.

3. Самые известные скоринговые функции ошибки: logloss, Brier score, exploss, они порождают информационные меры, которые используются в критериях расщепления: энтропийный, Джини, «корень из Джини».

Спасибо за внимание к книге!
Замечания по содержанию, замеченные ошибки и неточности можно написать в телеграм-чате
https://t.me/Dyakonovsbook